

```
In [71]: # first we are going to import necessary library
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```

if os.environ.get( 'RUNTIME_ENV_LOCATION_TYPE' ) == 'external' :
    endpoint_517c4add7ff840f8915f46a7a0f89295 = 'https://s3.ap.cloud-object-storage.appdomain.cloud'
else:
    endpoint_517c4add7ff840f8915f46a7a0f89295 = 'https://s3.private.ap.cloud-object-storage.appdomain.cloud'

client_517c4add7ff840f8915f46a7a0f89295 = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='jrUiIrk68Cc8AeuNY0kkYbwC-HP2b_RQ2y905E3I6Wvu',
    ibm_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url=endpoint_517c4add7ff840f8915f46a7a0f89295)

body = client_517c4add7ff840f8915f46a7a0f89295.get_object(Bucket='pythonprojectfordatascience-donotdelete-pr-i4fyreja3rirt',Key='Sacramentorealestatetra
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_data_1 = pd.read_csv(body)
df_data_1.head()

# call the data

```

Out[2]:

	street	city	zip	state	beds	baths	sq_ft	type	sale_date	price	latitude	longitude
0	3526 HIGH ST	SACRAMENTO	95838	CA	2	1	836	Residential	Wed May 21 00:00:00 EDT 2008	59222	38.631913	-121.434879
1	51 OMAHA CT	SACRAMENTO	95823	CA	3	1	1167	Residential	Wed May 21 00:00:00 EDT 2008	68212	38.478902	-121.431028
2	2796 BRANCH ST	SACRAMENTO	95815	CA	2	1	796	Residential	Wed May 21 00:00:00 EDT 2008	68880	38.618305	-121.443839
3	2805 JANETTE WAY	SACRAMENTO	95815	CA	2	1	852	Residential	Wed May 21 00:00:00 EDT 2008	69307	38.616835	-121.439146
4	6001 MCMAHON DR	SACRAMENTO	95824	CA	2	1	797	Residential	Wed May 21 00:00:00 EDT 2008	81900	38.519470	-121.435768

```
In [3]: # Linear Regression between beds and baths
x = df_data_1['beds'].values.reshape(-1,1)
y = df_data_1['baths'].values.reshape(-1,1)

In [4]: # call LinearRegression
linreg = LinearRegression()

In [5]: #call training dataset and test dataset
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1, random_state = 10)

In [6]: # fit the machine learning
linreg.fit(x_train,y_train)

Out[6]: LinearRegression()

In [7]: linregpredict = linreg.predict(x_test)

In [8]: # finding for accuracy score between y_test data set and y predict dataset
linreg.score(x_test, y_test )

Out[8]: 0.6368400546636221

In [9]: # finding regression coefficient
linreg.coef_

Out[9]: array([[0.57564847]])

In [10]: #finding regression intercept
linreg.intercept_

Out[10]: array([0.103917])
```



```
In [11]: #Plot the result
plot = plt.figure(figsize=[10,10])
plt.scatter(x_test, y_test)
plt.plot(x_test, linregpredict)
plt.xlabel('beds_test')
plt.ylabel('bath_tes')
plt.show()
```

